R & D STATUS REPORT

ARPA ORDER NO:	A418
PROGRAM CODE NO:	DO-C9
CONTRACTOR:	David Sarnoff Research Center
CONTRACT NO.:	N00014-93-C-0202
CONTRACT AMOUNT:	$676,870
EFFECTIVE DATE OF CONTRACT:	18 August 1993
EXPIRATION DATE OF CONTRACT:	17 August 1996
PRINCIPAL INVESTIGATOR:	John C. Pearson (609-734-2385)
TECHNICAL CONTRIBUTORS:	John Pearson, Paul Sajda and Clay Spence
SHORT TITLE:	Hybrid Pyramid / Neural Network Vision System
REPORTING PERIOD:	12/1/94 to 2/28/95

1 The views and conclusions contained in this document are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of the Defense Advance Research Projects Agency of the U.S. Government.
2 Ownership in Patent Data included herein is retained by the contractor/subcontractor pursuant to FAR 52.227-12.

Description of Progress:

A Learned Pattern Tree for a Toy Problem

To study the learning of pattern trees with neural nets, an artificial problem was constructed, in which the objects to be found and some potential false positives each have component patterns. To make the problem non-trivial, each positive has two component patterns, one each of two types, while each potential false positive has two components, both of the same type. Thus the pattern tree must detect both types of components for an object to be a positive. If only one type is detected, about half of the potential false-positives will be screened out . The positive and potential false-positives are 18-by-11 pixel rectangles with a value of 128, on a background with a value of 64. The component patterns are three-by-three x and + patterns whose pixels have the value 192 (Fig. 1).

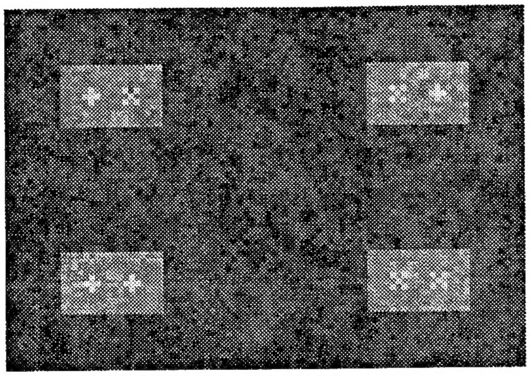

FIGURE 1. Examples of Positives and Potential False Positives. Each positive has both an "x" and a "+" pattern. The upper two rectangles are positive objects, while the lower two are negatives.

A background of noise is added to make identification from residual features at low resolution difficult. The noise is Gaussian white noise, added to the zeroth, first, and second levels of a Laplacian pyramid, from which a full-resolution noise image is constructed in the usual way. This gives significant noise at the pyramid levels which will be used in the pattern tree.

The objects, positive and potential false-positive, are arranged in a ten-by-twenty array, with one-hundred positives in the upper half of the image and one hundred negatives in the lower half. The horizontal and vertical spacings between the objects are twice their width and height, respectively, and they are spaced this far from the borders of the image, as well (Fig. 2).

A three-level Gaussian pyramid of the training image was made, and each neural net received as input a five-by-five window of pixels from the appropriate pyramid level. The training objective function is a sum over the positive "blobs" of the minimum of the cross-entropy errors at the positions in the blob, plus a sum over negative points of the cross-entropy error at those points. A weight-decay term was added, and roughly optimized over the regularization constant.

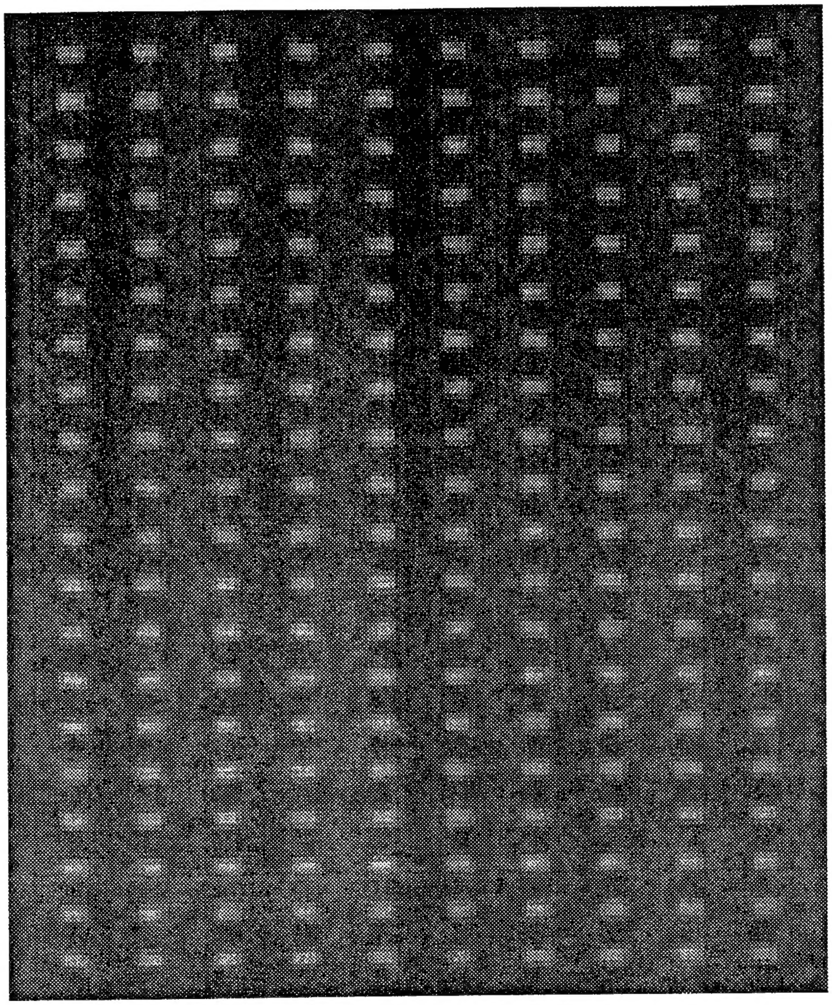

FIGURE 2. Second Level of Gaussian Pyramid of Training Image.

We call those networks which were trained with inputs from the image "component" networks, and those networks with inputs from the component networks “integration” networks. We further refer to them by the pyramid level at which each is used to search, level 0 being the full-size image and level 2 having one-quarter of the linear extent of the full-size image.

The level 2 component networks were trained on points from the entire image. Curiously, the nets did not learn to simply detect bright pixels, but something else which really distinguishes between the positives and negatives, although very poorly. On the test image, the best net can detect all 100 positives and miss two of the 100 negative rectangles, if the threshold is carefully adjusted. The nets respond only at a point (or two), apparently near the edge of each rectangle, possibly the corner. The exact position where a network goes high is not always on the feature being detected, which is allowed by the extended input window and the objective function. In any case, the position at which a network's output goes high is at a fixed offset from the feature. A net with a single hidden unit was optimal, but not much better than no hidden units.

The level 1 component net was trained on regions identified by the level 2 net. Since the level 2 net typically only went high at a pixel for each object, the level 1 net was trained in a rectangle around each such pixel, with the rectangle having a size and offset so it would cover the typical object. This seems reasonable for training on real objects, since we will know the regions of real objects also. The level 1 nets came out pretty much the same as level 2. The optimal net had two hidden layers, each with a single unit.

Since the level-1 net didn't do any better than the level-2 net, the level-0 net was trained in the same regions as the level-1 net, except of course at higher resolution. Twenty random starts of nets with no hidden layers resulted in one network which responded to the x patterns, and two which responded to the + patterns. The other seventeen networks all responded to corners of the object rectangles, and so were incapable of distinguishing positives from false positives. Although one of the + nets and the x net had lower than average errors, the other + net had a higher than average error.

The integration nets were trained in the blobs chosen by the level-2 component net. The inputs to the level-1 integration net were pixel values from five-by-five windows in the output images of three networks: the x level-0 net, the + level-0 net with lower error, and the best-performing level-1 component net. Thus there are seventy-five inputs. Fifty nets with no hidden units were trained, each starting from a different randomly-chosen weight vector. Twenty-four of these learned to respond to the upper-right corner (as did the level-1 component network which provides input), eleven responded to x patterns, nine responded to + patterns, two learned to correctly detect the positives and reject the potential false positives, and four had strange responses that didn't easily fit these categories.

The inputs to the level-2 integration nets were pixel values from five-by-five windows in the output images of three networks: the x level-0 net, the + level-0 net with lower error, and the best-performing level-2 component net. Thus there are 75 inputs. Ten nets with no hidden units were trained, and all have very low errors. All have strong responses at the desired objects. On test data, the threshold can be set so that six of the ten detect all of the desired objects and no false positives. Of the remaining four, in order to detect all of the desired patterns, the threshold had to be set to give between one and three false-positives.

These results tell us several things, in spite of the extreme simplicity of the problem:

1. Training several nets from different randomly chosen weight vectors can be an effective, if crude, way to generate network detectors for different features.

2. Networks can learn to detect unique features which nevertheless do not improve the pattern tree's ability to detect the objects of interest. At the second level, for example, different nets learned different corners, or parts of edges near corners, yet any one of the networks already achieved the best available detection performance at level 2. The others added nothing. This would not be the case for problems with occlusion. It may also happen that features which do not improve

detection performance have sub-features which do improve detection performance.

3. Many nets may learn useless features even when useful features are present. This happened with the level-0 component nets.

4. Competitive learning which forces networks to go high at different positions in the image may fail to force the nets to detect different features. With the five-by-five input windows, different nets could detect the same feature with their input windows centered on different points. This would not be the case if the detection of the feature required input from the entire window.

5. Given detections of features at one resolution, it may not be clear where the net at the next higher resolution should be trained. For entire objects, we can probably measure the mean or maximum extent of the objects around the pixels at which the network's output went high, and use similar regions around both true and false positives. However, to train nets to detect sub-features of lower-resolution features, there may not be a well-defined way to choose a region for training.

6. Cross-validation error does not necessarily distinguish between useful and useless features. The deficiencies of our objective function are making themselves felt here.

7. The outputs of the component networks make very good features for the integration networks, as was hoped.

The difficulties foreseen before trying this toy problem were mainly in training networks to each detect one feature, and to get each to detect a different feature. The first of these did not seem to be a problem with the current toy problem, but the desired objects of this problem have features which are present in all positive examples so that one feature sufficed to detect any positive. We have tried to detect different features by simply training from many different random restarts. This is not too expensive if the networks are simple, e.g., if they have no hidden units. Unfortunately, the experience with the level-0 component network (point 3, above) shows that features can be missed this way if too few restarts are tried.

We have begun investigating another artificial problem with greater variability in the objects' appearance, to see how the experiences above change with the added complexity. The objects in the current toy problem are very rigidly defined. Varying the brightness levels, in particular allowing them to be either brighter or darker than the surroundings will make the problem significantly harder. In addition, the desired objects in the new artificial problem have features which are not always present, in order to investigate potential problems with a single network learning multiple features.

Training neural networks with uncertain target positions

We developed an artificial problem to demonstrate the training algorithm we have developed for objects with uncertain positions. The objective function we used is the second of the two discussed in the last quarterly report. At the minimum of this function, the net is maximally likely on the training data to produce at least one positive response within each positive region, and negative responses at all locations outside of the positive regions. The error function is

$$E = -\sum_{x \in \text{Negatives}} \log(1 - y(\mathbf{f}(x))) - \sum_{i \in \text{Positives}} \log\left[1 - \prod_{x \in i}(1 - y(\mathbf{f}(x)))\right] \qquad \text{(EQ 1)}$$

in which $\mathbf{f}(x)$ is the input feature vector at position x, and $y(\mathbf{f})$ is the network output given $\mathbf{f}$.

The artificial problem is shown in Figure 3. Each of the objects to be found is a single pixel with a value of one. For the sake of clarity, they are arranged in a ten-by-ten grid. The network has no information about position, so it cannot use this fact to solve the problem. Each background pixel has a value of one-half or zero randomly assigned with equal probability (Fig. 3a). The desired outputs were specified incorrectly (most of the time) by placing the desired output at a randomly-chosen position within a three-by-three-pixel square centered around the correct position (Fig. 3b). A "network" consisting of a single neuron was used to search for the objects. The inputs to the neuron are the nine pixel values in a three-by-three window.

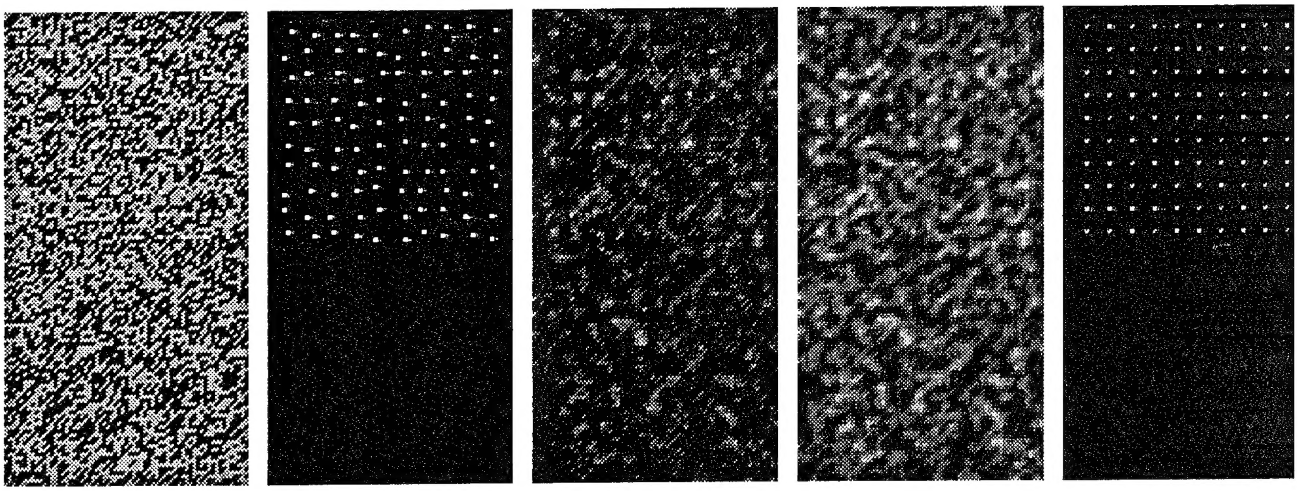

FIGURE 3. Example Problem. a) Input image. b) Mostly incorrect desired outputs. c) and d) Outputs of network trained with conventional methods (see text for details). e) Output of network trained with uncertain-position objective function. a) and b) are the training image, while c), d), and e) are outputs for a test image. The training and test images have objects in the same locations, but different background noise.

Figure 3c is the output of a network trained conventionally with the cross-entropy error function, assuming the positions in the desired output were correct. The pixel values of this image have been scaled, since the neuron produced a maximum output of only about 0.17.

Figure 3d is the output of a neuron trained conventionally with the cross-entropy error and a desired output which is one everywhere in three-by-three windows around the (possibly) incorrectly-specified object positions. This, too, has been scaled, but its maximum output was about 0.62.

Figure 3e is the output of a neuron trained using the uncertain-position objective function, in which case the net was trained to be likely to produce a detection within a three-by-three window around each of the incorrectly specified positions. This image has not been scaled.

As is apparent from Figure 3, the uncertain-position objective function produces superior results. It should be noted that a more complex net trained with a conventional method might work better than the conventionally-trained single-neuron networks used here. It would, of course, be more difficult to train. It is still true that a single neuron is capable of performing the task, but the usual training methods cannot teach it to do so.

Statistical Analysis of Mammography Data

Previously we have reported using our Hierarchical Neural Network (HNN) architecture for the problem of detecting microcalcifications in digital mammograms. We compared three networks architectures, showing that the HNN had significantly better accuracy than non-hierarchical architectures. Recently, we have compared the HNN with other detection/classification techniques, for example standard linear discriminants. In addition, we have begun to investigate different learning algorithms (e.g. Levenberg-Marquardt).

Comparison with a linear discriminant

We computed a linear discriminant for the microcalcification problem and compared its accuracy, through ROC analysis, to our HNN architecture and the non-hierarchical neural network architectures we have considered previously. Three different linear discriminant architectures were considered. One linear discriminant operated on features extracted from low resolution (LD, level 3) and another operated on features at high resolution (LD, level 3). A third discriminant was constructed by having the output of the low resolution (LD level 3) detector serve as additional input to the LD level 2 detector. This mimicked the hierarchical structure of the HNN. The table below summarizes the results, comparing the sensitivity of the two non-hierarchical neural networks, the three linear discriminants, and the HNN (sensitivity, A_g, is defined as the area under the ROC curve;

Detector	**Sensitivity (A_g)**
HNN	0.834
NN (level 3)	0.652

NN (level 2 & 3)	0.710
LD (level 3)	0.767
LD (level 2)	0.778
HLD	0.788

We first note that the HNN detector has the highest sensitivity of the six detectors. Secondly, the linear discriminant detectors all have higher sensitivities than the non-hierarchical neural network detectors, implying the increased accuracy of the HNN system must be attributed to its architecture, not to the fact that it simply uses neural networks. Though the hierarchical linear discriminant has the highest sensitivity amount the 3 LD detectors, its sensitivity is still far below the HNN. This implies that the key feature accounting for the high sensitivity of the HNN is that information is hierarchically propagated from the *hidden units.*

Learning algorithms

We have begun to look at variations in the learning algorithm to see if the accuracy of the HNN detector is dependent upon our use of cross-entropy as an error function or sequential quadratic programming as an optimization method. Initial simulations suggest that other error functions and optimization routines would also suffice. In particular, minimizing mean squared error using the Levenberg-Marquardt (LM) optimization method seems to promise results similar to SQP. However, we note that critical to the learning algorithm is a regularization term (e.g. weight decay). For instance, we have found that LM offers excellent results on the training set put very poor generalization on the test set. We are currently incorporating a regularization term into our mean square error minimization using LM and doing further testing.

Submission to Neural Networks, Special Issue on ATR

We submitted a paper entitled "Integrating Neural Networks with Image Pyramids to Learn Target Context" to the Special ATR issue of *Neural Networks*

Submission to International Conference on Image Processing (ICIP95)

We submitted the paper "A Hierarchical Neural Network Architecture that Learns Target Context: Applications to Digital Mammography" to ICIP95.

ICIP Paper submitted

A summary of a presentation titled "Training neural networks with uncertain object positions" was submitted for the 1995 International Conference on Image Processing.

NIPS Paper Submitted

A paper entitled "Coarse-to-Fine Image Search Using Neural Networks" was submitted for inclusion in *Advances in Neural Information Processing Systems 7,* the proceedings of the Neural Information Processing Systems Conference in Denver, CO, on November 30.

Software delivered to the Joint Warfare Analysis Center

At the request of the Joint Warfare Analysis Center, we delivered software to train a hybrid pyramid/neural network system to perform coarse-to-fine image search, including the possibility of exploiting context.

Mercury Computers

We have initiated talks with Mercury Computers regarding possible commercial applications of our Neural Network/Pyramid Software to problems in ATR and biomedical imaging. Mercury is a manufacturer of high-end parallel computer hardware, supplying platforms for both ATR and biomedical image processing. A potential partnership might include porting out neural network/pyramid algorithms onto their computer hardware.

Biomedical Applications (MRI)

We have been collaborating with Dr. Mitch Schnall, Director of MRI at the Hospital of the University of Pennsylvania. Dr. Schnall is a world leader in high resolution (180 micron) MRI and strongly advocates the use of these techniques for breast cancer screening, believing that the high resolution can pick up structural features absent in mammograms. However, the techniques he has been developing are very new, and it is unclear to him what features/characteristic/correlations in the high resolution MRI are cues to cancer. He believes our image processing/neural net techniques can serve as tools for helping him analyze this imagery.

Summary of Substantive Information Derived from Special Events:

None.

Problems Encountered and/or Anticipated:

None

Action Required by the Government:

The most recently scheduled funding increment has not occurred.

Financial Status

1. Amount currently provided on contract:	$225,740
2. Expenditures and commitments to date:	$319,762
3. Funds required to complete work:	$451,130

OFFICE OF THE UNDER SECRETARY OF DEFENSE (ACQUISITION)
DEFENSE TECHNICAL INFORMATION CENTER
CAMERON STATION
ALEXANDRIA, VIRGINIA 22304-6145

IN REPLY REFER TO DTIC-OCC

SUBJECT: Distribution Statements on Technical Documents

TO:
OFFICE OF NAVAL RESEARCH
CORPORATE PROGRAMS DIVISION
ONR 353
800 NORTH QUINCY STREET
ARLINGTON, VA 22217-5660

19951024 090

1. Reference: DoD Directive 5230.24, Distribution Statements on Technical Documents, 18 Mar 87.

2. The Defense Technical Information Center received the enclosed report (referenced below) which is not marked in accordance with the above reference.

QUARTERLY STATUS REPORT #6
N00014-93-C-0202
TITLE: HYBRID PYRAMID/NEURAL
NETWORK VISION SYSTEM

3. We request the appropriate distribution statement be assigned and the report returned to DTIC within 5 working days.

4. Approved distribution statements are listed on the reverse of this letter. If you have any questions regarding these statements, call DTIC's Cataloging Branch, (703) 274-6837.

FOR THE ADMINISTRATOR:

1 Encl

GOPALAKRISHNAN NAIR
Chief, Cataloging Branch

DISTRIBUTION STATEMENT A:

APPROVED FOR PUBLIC RELEASE: DISTRIBUTION IS UNLIMITED

DISTRIBUTION STATEMENT B:

DISTRIBUTION AUTHORIZED TO U.S. GOVERNMENT AGENCIES ONLY; (Indicate Reason and Date Below). OTHER REQUESTS FOR THIS DOCUMENT SHALL BE REFERRED TO (Indicate Controlling DoD Office Below).

DISTRIBUTION STATEMENT C:

DISTRIBUTION AUTHORIZED TO U.S. GOVERNMENT AGENCIES AND THEIR CONTRACTORS; (Indicate Reason and Date Below). OTHER REQUESTS FOR THIS DOCUMENT SHALL BE REFERRED TO (Indicate Controlling DoD Office Below).

DISTRIBUTION STATEMENT D:

DISTRIBUTION AUTHORIZED TO DOD AND U.S. DOD CONTRACTORS ONLY; (Indicate Reason and Date Below). OTHER REQUESTS SHALL BE REFERRED TO (Indicate Controlling DoD Office Below).

DISTRIBUTION STATEMENT E:

DISTRIBUTION AUTHORIZED TO DOD COMPONENTS ONLY; (Indicate Reason and Date Below). OTHER REQUESTS SHALL BE REFERRED TO (Indicate Controlling DoD Office Below).

DISTRIBUTION STATEMENT F:

FURTHER DISSEMINATION ONLY AS DIRECTED BY (Indicate Controlling DoD Office and Date Below) or HIGHER DOD AUTHORITY.

DISTRIBUTION STATEMENT X:

DISTRIBUTION AUTHORIZED TO U.S. GOVERNMENT AGENCIES AND PRIVATE INDIVIDUALS OR ENTERPRISES ELIGIBLE TO OBTAIN EXPORT-CONTROLLED TECHNICAL DATA IN ACCORDANCE WITH DOD DIRECTIVE 5230.25, WITHHOLDING OF UNCLASSIFIED TECHNICAL DATA FROM PUBLIC DISCLOSURE, 6 Nov 1984 (Indicate date of determination). CONTROLLING DOD OFFICE IS (Indicate Controlling DoD Office).

The cited documents has been reviewed by competent authority and the following distribution statement is hereby authorized.

A

(Statement)

OFFICE OF NAVAL RESEARCH
CORPORATE PROGRAMS DIVISION
ONR 353
800 NORTH QUINCY STREET
ARLINGTON, VA 22217-5660

(Controlling DoD Office Name)

(Reason)

(Controlling DoD Office Address, City, State, Zip)

DEBRA T. HUGHES
DEPUTY DIRECTOR
CORPORATE PROGRAMS OFFICE

19 SEP 1995

(Signature & Typed Name)

(Assigning Office)

(Date Statement Assigned)